

# Twitteranalyser

## DOKUMENTATION

621.252 PR Software Engineering II  
LV-Leiter: Dipl.Ing. Dr. Heinz Pozewaunig  
SS 2014

---

## **INHALTSVERZEICHNIS**

Dokumentation .....	1
Inhaltsverzeichnis .....	2
1. Einleitung .....	3
2. Projektorganisation.....	4
3. Zeiterfassung.....	5
3.1 Sprintverlauf.....	6
3.2 Stundenübersicht Sprints.....	7
4. Product Backlog .....	8
4.1. SOLL Product Backlog .....	8
4.2. IST Product Backlog .....	9
5. Kosten .....	10
6. Qualitätsplan.....	11
7. Technische Produktbeschreibung.....	11
7.1 Architektur .....	11
7.2 Controllers .....	12
7.3 Views .....	12
7.4 Apimanagement.....	12
7.5 Sprachen.....	12
8. Risikoanalyse.....	13
9. Anhang.....	14

---

## 1. EINLEITUNG

Im Rahmen der Lehrveranstaltung ‚Software Engineering II‘ wurde eine Twitteranalyse entwickelt, wobei als Softwareentwicklungsmodell *Scrum* Anwendung fand. Das Projekt, an dem acht Personen gearbeitet haben, wurde in drei Monaten, vom 18. März bis zum 18. Juni 2014 realisiert. Genaue Angaben zu den Kosten für das Produkt finden sich in Kapitel 5.

Wie beim Kundengespräch am 18.03.2014 von Ihnen erwähnt, sollte es möglich sein, dass ein Mitarbeiter in der Marketingabteilung ein Stimmungsbild einer bestimmten Veranstaltung durch die Sammlung von ausgewählten Tweets abrufen kann. Der Mitarbeiter sollte im Programm eine Veranstaltung anlegen, eine Veranstaltung auswählen, löschen, ändern oder speichern können. Es soll ein Zeitraum definiert werden können, bis wann die Tweets ausgewählt werden sollen. Die so erhaltene Sammlung soll analysiert, auf einer Landkarte dann die positiven und negativen Stimmungen angezeigt werden können und über die firmeninterne Analyseseite abrufbar sein.

Diese Anforderungen wurden alle in das Projekt aufgenommen, in fünf Sprints aufgeteilt und schließlich erfolgreich erledigt.

---

## **2. PROJEKTORGANISATION**

Das Scrum-Entwicklungsteam besteht aus acht Mitgliedern, deren Aufgabenbereiche in folgender Tabelle ersichtlich werden.

Name	Stärken	Schwächen	Aufgaben
Kesselbacher Max			Benutzer Dokumentation, User Tests
Kammel Johannes	Entwicklung, Projektmanagement, Logik	Dokumentation, Präsentation	Entwicklung, Logik, Datenbank, GUI
Krumpholz Maximilian			Unit Tests, Projektmanagement
Mertens Henrik			Unit Tests, Präsentation
Moser Manfred	Programmieren, Geduld, selbständiges Arbeiten, Arbeitstempo		Entwicklung, Logik, Datenbank, GUI
Mößlacher Corinna			Techn.Dokumentation, Datenbank, GUI
Ressmann Andreas			Projektmanagement, SetUp, Datenbank, GUI
Steinkellner Rosemarie	Dokumentation	Programmieren	Dokumentation, Kalkulation

---

### **3. ZEITERFASSUNG**

Im Weiteren wird der zeitliche Ablauf des Projekts beschrieben. Für jeden Sprint wurde ein Sprintziel definiert, das es zu erreichen galt. Nach dem ersten Sprint sollte bereits eine minimale Version des Twitteranalyser existieren. Dieses Sprintziel wurde nicht ganz erreicht, da ein Kollege überraschend das Team verlassen hatte. Daher konnte das Ergebnis aus Sprint I Veranstaltungen anlegen gemeinsam mit Veranstaltungen anzeigen und bearbeiten erst am Ende von Sprint II erfolgreich erreicht werden. Ziel von Sprint III und IV war es, Tweets anzeigen und zu einer Analyse zusammenfassen zu können. Außerdem sollte eine Übersicht der Tweets sowie eine multilinguale Ausgabe der Keywords möglich sein. Alle diese Ziele wurden erfolgreich umgesetzt. Im finalen Sprint V wurde das Ziel bereits angelegte Filter löschen können und die mögliche Darstellung der Analyse in Diagrammen ebenfalls erfolgreich umgesetzt.

Die folgende Tabelle bietet eine Übersicht über den gesamten Projektverlauf. Die Aufgabenbereiche sind dabei den dafür zuständigen Personen zugeteilt. Die grauen Balken zeigen an, über welchen Zeitraum diese Tätigkeiten erledigt wurden.

### 3.1 Sprintverlauf

Aufgabe	Person	Sprint I		Sprint II		Sprint III		Sprint IV		Sprint V	
Technische Dokumentation	Mößlacher										
Benutzer Dokumentation	Kesselbacher										
Dokumentation	Steinkellner										
Unit Tests	Krumpholz										
	Mertens										
User Tests, User Manuel	Kesselbacher										
Multilingualität	Steinkellner										
	Mertens										
	Mößlacher										
Logik	Ressmann										
Design	Ressmann										
GUI	Ressmann										
	Kammel										
	Moser										
	Mößlacher										
Sentiment, Twitter	Kammel										
	Moser										
Datenbank	Moser										
	Kammel										
	Mößlacher										
	Ressmann										
Filter-anpassung	Kammel										
	Moser										
	Ressmann										
Darstellung Analyse	Kammel										
	Ressmann										
	Moser										

---

In der folgenden Tabelle finden sich die genauen Stundenanzahlen, die pro Sprint in das Projekt investiert wurden und die gesamte Stundenanzahl aller Mitarbeiter, die sich auf 678 Stunden beläuft. Die folgende Abbildung zeigt den Zeitaufwand pro Sprint in Prozent.

### 3.2 Stundenübersicht Sprints

<i>Sprint</i>	<i>Stunden</i>
Sprint I	183:00
Sprint II	165:30
Sprint III	105:30
Sprint IV	104:00
Sprint V	120:00
<b>Gesamt</b>	<b>678:00</b>

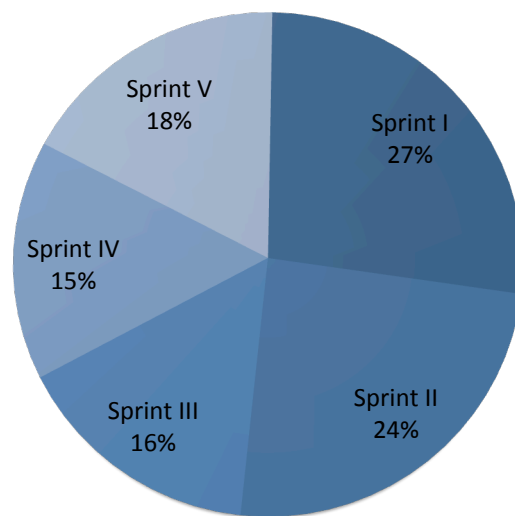


Tabelle 1: Zeitaufwand pro Sprint

---

## 4. *PRODUCT BACKLOG*

Die nachfolgende Abbildung stellt die ursprüngliche Liste mit Anforderungen, die nach dem Kundengespräch aufgestellt wurde, dar.

### 4.1. *SOLL Product Backlog*

Beschreibung	User Story Points	Priorität
Anlegen einer Veranstaltung	3	high
Webanwendung - Detailansicht	1	high
Parameteränderung	3	low
Auswertung	3	high
Filterobjekterstellung		high
Funktionalität: Löschen	2	medium
Filterobjektspeicherung		high
Funktionalität: Ausblenden	7	medium
Export	2	low
Darstellung	5	high
Benachrichtigung		low

Daraus wurde dann der folgende Product Backlog erstellt, der für die weiteren Sprints zur Abarbeitung bestimmt war. Den jeweiligen Anforderungen sind Aufgabenbereiche zugeteilt. Zudem ist angegeben, in welchem Sprint die Tätigkeit erfüllt wurde.



## 4.2. IST Product Backlog

ID	Sprint	Beschreibung	Priorität
T1a	1	Als <MM> will ich eine Veranstaltung anlegen, sodass das Sammeln der Tweets beginnt	100
T1b	2-1	Als <MM> will ich eine vorhandene Veranstaltung anzeigen	60
T1c	2-2	Als <MM> will ich eine vorhandene Veranstaltung bearbeiten	55
T2a	3-2	Als <MM> will ich das Tweets nach Anlegen einer Veranstaltung zur Verfügung haben, so dass ich später eine Sentimentanalyse fahren kann	99
T2b	3-4	Als <MM> will ich die gesammelten Tweets einer Veranstaltung alle anzeigen, sodass ich ein Übersicht bekomme	55
T3a	3-1	Als <MM> will ich eine positiv/negativ Sentiment Analyse der Tweets erhalten	40
T3b	4-1	Als <MM> will ich die vorhandenen Tweets einschränken, sodass nur relevante Tweets analysiert werden	30
T4a	5-1	Als <MM> will ich bereits angelegte Filter auch löschen können.	30
T4b	3-3	Als <MM> will ich eigene Keywords in beliebiger Sprache für die positiv / negativ Sentimentanalyse bestimmen können.	27
T4c	2-1	Als <MM> möchte ich eine Mindestanzahl von zu sammeln Tweets pro Veranstaltung angeben, damit ich bewerten kann, ob das Analyseergebnis sinnvoll ist.	26
T4d	4-3	Als <MM> will ich einzelene Tweets manuell löschen, sodass diese nicht wieder analysiert werden.	20
T4e	4-2	Als <MM> will ich einzelene Tweets manuell für einen Filter ausblenden, sodass diese nicht analysiert werden.	19
	5	Als <MM> will ich zu einer bestimmten Veranstaltung gehörende Daten als .csv exportieren.	15
		Als <MM> will ich verschiedene Darstellungen der analysierten Daten.	
	3-2		16
	1-5	Als <MM> will ich über Tweets informiert werden.	10

---

## 5. KOSTEN

Nachfolgend findet sich eine Aufstellung der Kosten für das Projekt (siehe auch die Aufwandsschätzung vom 05.05.2014):

Die Kostenschätzung mittels Function-Point Verfahren ergibt 54 UFP und 47,52 AFP.

Die Kostenschätzung mittels Application-Point Verfahren beträgt bei einem System mit 12 Screens, 4 Berichten mittlerer Komplexität, 6 3GL Modulen und ca. 25 % Wiederverwendung, eine gute Umgebung vorausgesetzt und wenn wir schätzen, dass die Mitarbeiter mittlere Erfahrung haben 104 AP's und 17 NAP's = 6 Personenmonate

Der Durchschnitt der beiden Varianten ergibt nun 13 Personenmonate.

Die Gesamtkosten errechnen sich wie folgt:

Stundensatz von 90 Euro x 678 Stunden .....	61.020,00 €
<hr/>	
Nettorechnungsbetrag.....	61.020,00 €
+ 20 Prozent Umsatzsteuer .....	12.204,00 €
<hr/>	
<b>Bruttorechnungsbetrag .....</b>	<b>73.224,00 €</b>
<hr/>	

---

## **6. QUALITÄTSPLAN**

Schon während der Entwicklung des Software-Produkts wurden regelmäßig Unittests durchgeführt, um eine hohe Qualität zu gewährleisten. Zudem wurde das Produkt von Mitarbeitern und externen Personen kontinuierlich auf Benutzerfreundlichkeit getestet. Im Anhang befinden sich Links zu den Ergebnissen dieser Tests. Außerdem ist dort eine Verlinkung zu den Code-Metriken zu finden.

## **7. TECHNISCHE PRODUKTDESCHEIBUNG**

Im Folgenden wird auf die technische Realisierung der Software näher eingegangen. Für nähere Details siehe auch die technische Dokumentation, die sich auf der CD befindet.

Die Software wurde mittels der objektorientierten Programmier- und Skriptsprachen PHP und Java realisiert.

Die Struktur des Projektes ist eine klassische MVC Architektur, realisiert mit Zend Framework (Version 1).

### ***7.1 Architektur***

Der Beginn der Anwendung ruft die Index.php Datei auf welche den Zend Bootloader für die primäre Initialisierung des Projekts ladet. Alle Anfragen, die über die Website empfangen werden, werden an die entsprechende Steuerung durch das Zend Framework weitergeleitet. Die Steuerung wird durch die Unterordner der Anwendung bestimmt. Alle Hauptmenüpunkte haben eine entsprechende Steuerung. Der Standard Controller ist der Index Controller. Alle Menüunterpunkte sind Aktionen im entsprechenden Controller. Jeder Klick auf einen Unterpunkt ruft eine andere Aktion in der Steuerung auf. Die Index Action wird aufgerufen, wenn der Controller aufgerufen wird. Die Anwendung nutzt Twitter-Bootstrap. Dies ist für die Gestaltung der grafischen Benutzeroberfläche und der Präsentation (z.B. der Datumsauswahl) verantwortlich.

---

## 7.2 Controllers

Die Controller Logik ist in fünf Controller aufgeteilt:

Index Controller (Haupt Controller), API Controller, Error Controller, Map Controller und Statistics Controller

## 7.3 Views

Die phtml-Dateien in "Views" sind für die Darstellung der Web-Seite eingesetzt. Sie enthalten neben den HTML-Code auch PHP-Code für die dynamischen Aspekte der Seite. Alle Nutzer-Interaktionen rufen eine Aktion im Controller hervor, zu der die View gehört. Für die Views "Analyse", "Event" und "Tweets" ist es der API-Controller. Die View "Ereignis" erzeugt eine Liste als Übersicht aller derzeit bestehenden Veranstaltungen. Für die Ansicht der Karte und der Statistik werden Google Maps und Google Visualisation für die Darstellung der Karte und Diagramme verwendet.

## 7.4 Apimanagement

Die Dialoge sind durch die Datei "apimanagement.js" verwaltet. Alle Dialoge der Web-Anwendung werden unter Verwendung von EJS-Code generiert, der geladen und dynamisch über JQuery angezeigt wird. Das API Management sendet JSON API-Anfragen an den Controller. Der Controller sendet Antworten oder Validierungsfehler.

## 7.5 Sprachen

Die CSV-Dateien enthalten die Texte, die auf der Webseite angezeigt werden.

## 7.6 Models

Für alle Datenbanktabellen, die relevant für die Ansichten sind, existiert ein entsprechendes Modell.

---

## **8. RISIKOANALYSE**

In der nachfolgenden Tabelle werden mögliche Risiken, die Wahrscheinlichkeit ihres Eintritts und Pläne zur Vermeidung des jeweiligen Risikos aufgelistet. Die Auswertung, der Durchschnitt von Wahrscheinlichkeit und Schweregrad, wird von 2 = geringes Risiko bis 4 = hohes Risiko definiert.

<b><i>Risiko</i></b>	<b><i>Wahrscheinlichkeit</i></b>	<b><i>Schweregrad</i></b>	<b><i>Auswertung</i></b>	<b><i>Gegenmaßnahmen</i></b>
Personalausfall	Möglich	Moderat	2	Konfliktarbeit, Motivation der Mitarbeiter
Unrealistische Termin- und Budgetplanung	Wahrscheinlich	Hoch	4	Genaue Aufwands- und Kostenschätzung
Entwicklung falscher Funktionalitäten	Möglich	Hoch	3	Genaue Anforderungsanalyse, Kundengespräche protokollieren, stetiger Kundenkontakt,
Entwicklung einer falschen Benutzeroberfläche	Möglich	Hoch	3	Kundengespräche protokollieren, stetiger Kundenkontakt, genaue Anforderungsanalyse, Usability-Tests
Anforderungen ändern sich kontinuierlich	Möglich	Moderat	2	Leicht änderbare Software entwickeln
Nichterfüllen der gewünschten Anforderungen	Möglich	Hoch	3	Genaue Zeitplanung, gute Arbeitsaufteilung
Mangelnde Echtzeit-Performance	Möglich	Hoch	3	Kontinuierliche Tests, Messung und Bewertung
Datenverlust	Möglich	Schwerwiegend	4	Backups, Versionsverwaltung mit Git

---

## **9. ANHANG**

Auf der CD befinden sich neben der Dokumentation auch die Testdokumente und das Dokument zu den Codemetriken. Natürlich beinhaltet die CD auch die Software für die Twitteranalyse an sich.